

Requested Patent: JP2000057709A

Title:

SIGNAL PROCESSING APPARATUS AND METHOD, AND DATA
RECORDING/REPRODUCING APPARATUS USING THE SAME ;

Abstracted Patent: US6493846 ;

Publication Date: 2002-12-10 ;

Inventor(s): KONDO MASAHARU (JP); MITA SEIICHI (JP) ;

Applicant(s): HITACHI LTD (JP) ;

Application Number: US19990323703 19990601 ;

Priority Number(s): JP19980154178 19980603; JP19990119191 19990427 ;

IPC Classification: H03M13/03 ;

Equivalents: ;

ABSTRACT:

A signal processing apparatus capable of reducing burst error generation, and a highly reliable data recording/reproducing apparatus using this signal processing apparatus. This signal processing apparatus has a simple error detection/correction circuit provided just before a modulated code demodulator, thereby correcting error of a pattern easy to occur in a maximum likelihood decoder. The simple error detection/correction circuit is an error detection/correction circuit using a linear error correction code, for example, an error correction code (CRCC) formed of a cyclic code. Thus the number of burst errors after the modulated code demodulator can be decreased

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号
特開2000-57709
(P2000-57709A)

(43)公開日 平成12年2月25日(2000.2.25)

(51)Int.Cl. ⁷	識別記号	F I	テーマコード(参考)
G 1 1 B 20/18	5 3 4	G 1 1 B 20/18	5 3 4 A
	5 7 0		5 7 0 F
20/14	3 4 1	20/14	3 4 1 A
H 0 3 M 13/00		H 0 3 M 13/00	

審査請求 未請求 請求項の数11 O L (全 11 頁)

(21)出願番号 特願平11-119191

(22)出願日 平成11年4月27日(1999.4.27)

(31)優先権主張番号 特願平10-154178

(32)優先日 平成10年6月3日(1998.6.3)

(33)優先権主張国 日本(J P)

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 近藤 昌晴

神奈川県小田原市国府津2880番地 株式会
社日立製作所ストレージシステム事業部内

(72)発明者 三田 誠一

神奈川県小田原市国府津2880番地 株式会
社日立製作所ストレージシステム事業部内

(74)代理人 100075096

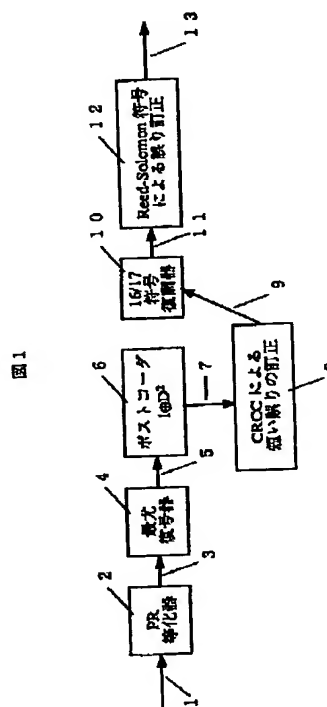
弁理士 作田 康夫

(54)【発明の名称】 磁気ディスク誤り訂正方法及び装置

(57)【要約】

【課題】本発明の目的は、変調符号による0連続長制限を崩すこと無く、冗長度の小さい、しかも構成の容易な誤り訂正符号と、それを用いた信号処理方式を提供することにある。

【解決手段】変調符号化後にCRCCの冗長ビットを付加し、CRCCの誤りシンδροームを利用することで高頻度で発生する短い誤りをデータ内から除去することにより、変調符号復調器の直前に最尤復号器で発生し易い短い誤りを抑制することができ、その結果、変調符号復調器による誤りの拡大を減少させることが可能となる。



【特許請求の範囲】

【請求項1】0連続長を制限する変調符号化されたデジタル情報の予め設定された長さのデータブロック毎に予め設定された数のビットを有する冗長ビットを付加する形式で符号化ブロックを構成する方法であって、冗長ビットの形式として、上位に0連続長制限ビットを有し、変調符号化されたデジタル情報列の予め設定された長さのデータブロックに該0連続長制限ビットを連結したビット列に対する巡回符号化冗長ビットが該0連続長制限ビットの後に付加されていることを特徴とする誤り訂正符号化方法。

【請求項2】0連続長を制限する変調符号化されたデジタル情報の予め設定された長さのデータブロック毎に予め設定された数のビットを有する冗長ビットを付加する形式で符号化ブロックを構成する方法であって、冗長ビットの形式として、下位に0連続長制限ビットを有し、変調符号化されたデジタル情報列の予め設定された長さのデータブロックに該0連続長制限ビットを連結したビット列に対する巡回符号化冗長ビットが該0連続長制限ビットの前に付加されていることを特徴とする誤り訂正符号化方法。

【請求項3】請求項1、又は請求項2に記載の誤り訂正符号化方法において、EPRML、EPRML及びこれらのレスポンス係数を変えたMEPRMLやMEPRMLの最尤復号器で発生する誤りパターンの中で予め選ばれたものが、変調符号化されたデジタル情報の予め設定された長さのデータブロックと該予め設定された数のビットを有する冗長ビットを合わせた符号化ブロックで発生したときに巡回符号による誤りシンδροームを用いることで誤り訂正が可能であることを特徴とする誤り訂正符号化方法。

【請求項4】請求項3に記載の誤り訂正符号化方法において、符号化ブロックに1か所或いは符号化ブロック2つにまたがる形で誤りが1か所発生したときに、巡回符号による誤りシンδροームを用いることで誤り訂正が可能であることを特徴とする誤り訂正符号化方法。

【請求項5】請求項4に記載の誤り訂正符号化方法において、トラップ復号法に誤りシンδροームのパターンと更新回数とのマッチングを併用して誤り位置と誤りパターンを算出する誤り訂正符号化方法。

【請求項6】請求項1、又は請求項2、又は請求項3、又は請求項4、又は請求項5に記載の誤り訂正符号化方法を用いた信号処理方式。

【請求項7】請求項1、又は請求項2、又は請求項3、又は請求項4、又は請求項5に記載の誤り訂正符号化方法、及び請求項6に記載の信号処理方式において、巡回符号の生成多項式として16進表記で0x72CDを用いた誤り訂正符号化方法。

【請求項8】請求項7に記載の誤り訂正符号化方法において、(0,6/6)の0連続長制限を満たすために該0連続

長制限ビットとして3ビットを持つような誤り訂正符号化方法。

【請求項9】請求項8に記載の誤り訂正符号化方法において、特に該0連続長制限ビットの最上位が「1」で固定されているような誤り訂正符号化方法。

【請求項10】最尤復号器から得られた尤度情報を用いて誤りイベントの発生を検出し、該符号化ブロックで検出した誤りイベント数が2以上か否かによって、符号化ブロック内の誤りの放置・訂正の動作を切り換えることを特徴とした請求項5、又は請求項7、又は請求項8、又は請求項9に記載の誤り訂正方式及び請求項6に記載の信号処理方式。

【請求項11】請求項1から請求項10迄に記載の誤り訂正符号化方法及び信号処理方式のうち少なくとも一つ以上を用いた回路及び磁気記録再生装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、短い誤りを訂正するための冗長ビット部0連続長制限付き符号化方法の構成に関する。

【0002】

【従来の技術】磁気ディスク装置への高記録密度化の要望はますます高まっており、これを支える記録再生系の信号処理技術も高記録密度化に対応してきた。

【0003】図2は従来のEPRMLにおけるデジタル情報の読み出し処理の流れを表わしたものである。ヘッドから読み出された信号1は、等化器2によって等化され、等化後の信号3から最尤復号器4によって、実際に記録されていた符号系列が推定される。推定された符号系列5は、ポストコード6を通して、16/17符号復調器10によって対応する情報系列11に復号される。対応する情報系列11は、Reed-Solomon符号による誤り検出・訂正12が行なわれる。

【0004】最尤復号器4における誤り系列は、正しい系列との距離が短いものが大半を占める。正しい系列との距離が短い誤り系列を調べるには、誤り流れ図を用いる。図3はEPRMLにおける正しい系列との距離が8以内の誤り流れ図を簡略化したものである。各状態(e_{t-3} e_{t-2} e_{t-1} e_t)において、 e_t は時刻tにおける誤りを示している。 e_t が0のときは、対応するビットが誤っていないことを、+のときはビット「0」が「1」に誤っていることを、-のときはビット「1」が「0」に誤っていることを表す。また、流れ図の枝に付けられた数字は、対応する誤りの遷移によって増加する正しい系列との距離を示す。図3より、EPRMLにおける最尤復号器4の誤り系列は、正しい系列との距離の短い順に $\pm(+++)$ (連続3ビット誤り)、 $\pm(+++)$ (連続4ビット以上の誤り)、 $\pm(++0+)$ の誤りであることわかる。実際の誤りの発生頻度は、正しい系列との距離の他に誤りの長さや雑音の相関が影響し、連続3ビット、1ビット、2ビッ

ト、5ビット、4ビットの順になる。但し、ここでの連続誤りは「0101…」が「1010…」に、或いはその逆方向に誤る形式になっている。

【0005】更に上位のPRMLを使ったり、パーシャルレスポンスの係数を少し動かたりしても、頻度順は、多少前後することはあるが誤りの傾向は変わらない。

【0006】

【発明が解決しようとする課題】最尤復号器4における変調符号上の短い誤りは、変調符号復調器10によってバースト誤りへと拡大される。16/17変調符号を例にとると、最悪で4バイトまで拡大する。これは、誤りが16/17符号の末尾で発生し、ポストコード6によって次のコードワードに誤りが伝搬する場合の最悪値に相当する。このような誤りの拡大は、結果としてReed-Solomon符号の訂正能力を低下させることになる。

【0007】

【課題を解決するための手段】上記課題を解決するための手段として、本発明を用いた信号処理方式を図1に示す。

【0008】変調符号復調器10の直前に簡単な誤り訂正8によって、最尤復号器4で発生し易いパターンの誤りのみ訂正する。これによって、変調符号復調器10の後のバースト誤りの数を減らすことができる。

【0009】線型誤り訂正符号とは、訂正対象とする誤りであれば、誤りを検出するためのパリティ値（以下、誤りシンδροーム）からその誤りが一意に求めることができるような誤り検出符号である、と言い換えることができる。fを誤り多項式から誤りシンδροームへの写像、 E_0 を訂正対象となる誤り多項式全体とすると、 $f|E_0$ （fの定義域を E_0 に制限した写像）は単射である（図4）。

【0010】誤り検出符号として、回路化が容易な巡回符号を用いる。巡回符号を使う場合、fは誤り多項式を巡回符号の生成多項式で割った剰余を求める操作に相当する。実際に誤りシンδροームを算出するには誤り検出器501を用いる。

【0011】誤り訂正のアルゴリズムは、トラップ復号法と呼ばれる巡回符号によるバースト誤り訂正の手法が知られているのでそれを用いる。これは、生成多項式g(x)に対し、 $x^e \equiv 1 \pmod{g(x)}$ であれば誤りシンδροームにxを掛けてg(x)で割った余りをとるのを繰り返すことで誤りパターンが出現することを利用した手法である。これより、少なくともg(x)はxを因数に持たないようなものを使う必要がある。

【0012】以下、巡回符号から構成された誤り訂正符号をCRCC (Cyclic Redundancy Check Code) と表記する。

【0013】以下、CRCCの構成について述べる。

【0014】 $f|E_0$ が単射であることから、巡回符号の冗長ビット数 p' と E_0 の要素数 $\#E_0$ は数1の不等式を満た

す。これに基づいて p' を決定する。

【0015】

【数1】 $2^{p'} \geq \#E_0$ …… (数1)

p' をできるだけ小さくするために、すなわち訂正対象となる誤り多項式を必要最小限に抑えるために、発生頻度の高い誤りがCRCCの符号化ブロックに1か所、或いは符号化ブロック2つにまたがる形で1か所発生した場合のみ誤りを訂正するようなCRCCを構成する。

【0016】実際に訂正を行なう誤りパターンは、最尤復号器4で起こる誤りを図1におけるポストコード6に通した後のものであることに注意する。また、2つのCRCCの符号化ブロックにまたがるときはその一部分になる。

【0017】しかし変調符号化されたビット列を一定間隔kビットずつ毎に区切ったデータブロックに誤り検出器501によって、算出された巡回符号の冗長ビットを単純に付加するのではその部分で変調符号による0連続長制限が大きく崩れる。巡回符号の冗長ビットが剰余であることから、冗長ビット全てが「0」になることがあり得るからである。

【0018】そこで、付加する冗長ビット数を $p(>p')$ とし、データブロックkビットに対応する冗長ビットから $p-p'$ ビットの0連続長制限ビットを計算する。この0連続長制限ビットを用いて巡回符号の冗長ビットを修正することで0連続長の制限を行なうことにする。このとき、0連続長制限ビットと更新後の巡回符号の冗長ビットの組が、CRCCの符号化ブロックの冗長ビットとなる。冗長ビットの修正は、0連続長制限を満たし、かつkビットのデータブロックに冗長ビットを付加した系列がg(x)を生成多項式とする巡回符号語になるように行なう。この冗長ビットの修正は、0連続長制限ビットが冗長ビットの上位にあっても下位にあっても、殆ど同じ手順で冗長ビットの修正を行なうことができる。

【0019】

【発明の実施の形態】以下、本発明に係る誤り訂正符号及び誤り訂正装置の実施例について、図5～10を参照しながら説明する。

【0020】一例として、16/17 (0,6/6) 変調符号32コードワード ($k=544$) 毎に、0連続長制限ビット込みの冗長ビットをpビット付加するような符号長 $544+p$ のCRCCで、EPRMLと組み合わせることを想定したものを構成する。以下、誤りパターンを表現する際、誤っていることを「e」、誤っていないことを「o」と表記する。

【0021】プリコードは $1/(1+D^2)$ 、ポストコードは $1+D^2$ とする。

【0022】0連続長制限ビットは冗長ビットの上位に置くように構成する。

【0023】EPRMLにおける最尤復号器4の誤りの発生頻度の上位を占めるのは、連続1～5ビット誤りであった。従って、訂正対象とする誤りパターンはポストコー

ダ6を通った後のパターンであるeoc、eeee、ecoc、ecoc、ecoc、ecoc、そしてこれらがCRCCの符号化ブロックでまたがるときに現われるパターンとすれば良い。 E_0 はこれらがCRCCの符号化ブロック中に1つ現れるのに相当

$$E_0 = \left\{ x^m \times (x^2 + 1) \mid m \geq 0, m+2 < 544+p \right\} \\ \cup \left\{ 1, x, x^{543+p}, x^{542+p} \right\} \\ \cup \left\{ x^m \times (x^3 + x^2 + x + 1) \mid m \geq 0, m+3 < 544+p \right\} \\ \cup \left\{ x+1, x^2+x+1, x^{542+p}(x+1), x^{541+p}(x^2+x+1) \right\} \\ \cup \left\{ x^m \times (x^4 + x^3 + x + 1) \mid m \geq 0, m+4 < 544+p \right\} \\ \cup \left\{ x^2+x, x^3+x^2+1, x^{541+p}(x+1), x^{540+p}(x^3+x+1) \right\} \dots\dots (数2) \\ \cup \left\{ x^m \times (x^5 + x^4 + x + 1) \mid m \geq 0, m+5 < 544+p \right\} \\ \cup \left\{ x^3+x^2, x^4+x^3+1, x^{540+p}(x+1), x^{539+p}(x^4+x+1) \right\} \\ \cup \left\{ x^m \times (x^6 + x^5 + x + 1) \mid m \geq 0, m+6 < 544+p \right\} \\ \cup \left\{ x^4+x^3, x^5+x^4+1, x^{539+p}(x+1), x^{538+p}(x^5+x+1) \right\}$$

【0025】誤りを訂正する際に誤訂正が起こり得る。誤訂正とは、訂正対象外の誤りに対応する誤りシンδροームが訂正対象になっている別の誤りに対応する誤りシンδροームと一致することで、訂正動作の結果として別の位置に誤りを増やす現象である。

【0026】従って、確率的に誤訂正を起こし難いことを誤り多項式を選ぶ基準とする。 E_0 の次に発生し易い誤りに対応する誤り多項式全体 E_1 を定める。 E_1 の要素の誤りシンδροームが、 E_0 の要素の誤りシンδροームのいづれにも一致しない、或いは一致したとしても問題が、極めて小さいときに誤訂正を起こし難いと見なす。

【0027】最尤復号器4において、連続1〜5ビット誤りの次に起こり易い誤りのパターンとして、残りの誤りパターンの中で距離が短いものである、連続6ビット以上の誤り、そして±(+00+-+)に対応するパターンeeeeoeeeまで考える。これは距離8の誤りで E_0 に含まれない誤りパターンに相当する。

【0028】0連続長制限ビットと巡回符号の冗長ビットとその前後における0連続長制限は、16/17変調符号のそれと同じ(0,6/6)にすることを目標とする。このとき、図6に示す通り、プリコードにビット列10000001が入力されたときの出力において±(+00+-+)の形の連続誤り(距離8)の長さは最大8まで起こり得ることがわかる。すなわち、 E_1 で想定する連続誤り長は最大8まで考慮すれば良い。

【0029】従って、長さ8までの連続誤りをポストコード6に通したパターン、eeeeoeeeをポストコード6に通したパターンecocococ、そしてそれらがCRCCの符号化ブロックでまたがるときに現われるパターン全てが $E_0 \cup E_1$ に含むように E_1 を定めてやれば良い。

【0030】次に冗長ビットについて検討を行なう。 $2^{12} < \#E_0 < 2^{13}$ より、冗長ビット数pは16/17変調符号の符号長である17にしておけば充分である。符号長は561(=17×32+17)ビットとなる。

する誤り多項式全体となる。これを数2に示す。

【0024】

【数2】

【0031】 $p=17$ のとき、0連続長制限ビット数 $s=p-p'$ を2以下にはできない。つまり、生成多項式として17-s次の多項式をどう選んだとしても、0連続長制限(0,6/6)を満たすように0連続長制限ビットを採ることができないような巡回符号の冗長ビットが存在する。よって、0連続長制限ビット数を3、巡回符号の冗長ビット数 p' を14とする。

【0032】冗長ビット数14の完全巡回符号の符号長は最大で $2^{14}-1$ である。以下、一例として生成多項式として符号長が最大となる原始多項式をとる。このとき、目的の符号は、符号長 $2^{14}-1$ の完全巡回符号を長さ561ビットに短縮したものになる。トラップ復号法を行なう前には、誤りシンδροームを切り捨てたビット長に相当する回数だけ誤りパターンのチェックを省いた空回りをさせる必要がある。

【0033】しかし、単純に誤り検出器501で切り捨てたビット長に相当する回数のビットシフトを行なうと遅延が大きくなる。そこで、誤りシンδροームを x^{13} , ..., $x, 1$ を基底とするベクトルと見なし、固定の正方行列を誤りシンδροームのベクトルにかけるという形式で前記切り捨てたビット長に相当する回数の更新に対応する操作を表現する。この演算形式であれば行列やベクトルの次数が、余程大きくならない限り遅延は気にならなくなる。

【0034】図7は、生成多項式とすることで最尤復号器4において発生した E_0 に属する誤りを訂正可能でかつ長さ6の連続誤りでは誤訂正を起こさないような14次の原始多項式と、 E_1 に属する誤りでの誤訂正の有無の一覧である。生成多項式 $g(x) = g_{14}x^{14} + g_{13}x^{13} + \dots + g_2x^2 + g_1x + g_0$ ($g_1=0$ 又は1、但し $g_0=g_{14}=1$)を $g_{14}2^{14} + g_{13}2^{13} + \dots + g_22^2 + g_12 + g_0$ の16進表記で表わしている。また、誤訂正によって増える誤りが、CRCCの冗長ビットに含まれる場合、冗長ビットが、取り除かれることによって、結果的に誤りを放っておいたのと等価にな

る。従って、このようなときは誤訂正と見なさないことにしている。図7の結果より、最尤復号器4における誤りパターンが連続8ビット以下の誤りであってもeeeeeeでであっても誤訂正を起こさない生成多項式は16進表記0x72CDのみである。

【0035】以下に述べるのは、生成多項式0x72CDにおけるCRCCの冗長ビットの計算の一例である。図8は説明における冗長ビットの構造を示す。

【0036】(1) 16/17符号32コードワードに対応する巡回符号の冗長ビットを誤り検出器501を使って算出

$$\begin{aligned} s_1 = & \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_9} \overline{c_4} \vee \overline{c_{13}} \overline{c_{11}} \overline{c_4} \vee \overline{c_{13}} \overline{c_{11}} \overline{c_4} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_6} \overline{c_4} \\ & \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_6} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_6} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_6} \overline{c_3} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_6} \overline{c_3} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_3} \\ & \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_3} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_4} \overline{c_3} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_3} \overline{c_0} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_6} \overline{c_3} \overline{c_0} \\ & \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_6} \overline{c_3} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_6} \overline{c_3} \overline{c_1} \overline{c_0} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_6} \overline{c_1} \overline{c_0} \vee \overline{c_{12}} \overline{c_{11}} \overline{c_3} \overline{c_1} \overline{c_0} \\ & \vee \overline{c_{13}} \overline{c_{12}} \overline{c_5} \vee \overline{c_{12}} \overline{c_3} \overline{c_4} \overline{c_1} \overline{c_0} \end{aligned}$$

..... (数4)

【数5】

【0040】

$$\begin{aligned} s_0 = & \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_6} \overline{c_3} \overline{c_0} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_3} \overline{c_0} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_3} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_6} \overline{c_3} \\ & \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_3} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_6} \overline{c_3} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_3} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_6} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_6} \overline{c_1} \\ & \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_6} \overline{c_1} \overline{c_0} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_4} \vee \overline{c_{13}} \overline{c_{11}} \overline{c_4} \vee \overline{c_{13}} \overline{c_{11}} \overline{c_4} \overline{c_3} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_4} \\ & \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_3} \vee \overline{c_{13}} \overline{c_{12}} \overline{c_{11}} \overline{c_{10}} \overline{c_3} \end{aligned}$$

..... (数5)

【0041】但し、上線は「0」と「1」の反転を、 \vee は論理和 (OR) を表わす。

【0042】(3) (s_2, s_1, s_0) を0連続長制限ビットとして、誤り検出器に s_2, s_1, s_0 の順に入力することによって0連続長制限ビットの後に付加する冗長ビットの残り $c_{13}, c_{12}, \dots, c_2, c_1, c_0$ を算出する。

【0043】この0連続長制限ビットによってCRCCの冗長ビットの前後における0連続長制限が(0,6/6)を満たしていることが確認できる。

【0044】数3、数4、数5を適切に変え、 $c_{13}, c_{12}, \dots, c_2, c_1, c_0$ に $(s_2 x^2 + s_1 x + s_0)g(x)$ を加えるようにすれば0連続長制限ビットが冗長ビットの下位にある場合に対応できる。

【0045】以上のように構成されたCRCCにおける訂正アルゴリズムを図9を用いて説明する。[]で囲まれている番号は図に示された番号に対応している。

【0046】以下、多項式計算において「mod $g(x)$ 」と書くのが冗長であることと $g(x)$ が原始多項式であることから、 $F_2[x]$ (0と1からなる体を係数とし、 x を不定元とする多項式全体) の $g(x)$ による剰余類をGalois体の元と見なし、剰余類の不定元 x を β と表わす。誤りシンδροームに β を掛けるというのは誤り検出器501にビット「0」を入力することで誤りシンδροームを更新することと相当する。

【0047】【901】誤り検出器501によってCRCCの符号化ブロック561ビットにおける誤りシンδροーム β を算出する。

する。

【0037】(2) (1)で求められた巡回符号の冗長ビットを $c_{13}, c_{12}, \dots, c_2, c_1, c_0$ とすると、0連続長制限ビット (s_2, s_1, s_0) を数3、数4、数5のようにおく。

【0038】

【数3】

$$s_1 = 1 \quad \dots \dots \text{(数3)}$$

【0039】

【数4】

【0048】【902】誤りの検出。 $r(\beta) = 0$ なら誤り無しと見なし7911のように訂正情報を設定する。

【0049】【903】空回り処理。 $r(\beta) \neq 0$ に定数 β $(2^{14}-1) - (561-14)$ をかけて $r(\beta)$ を更新する。この操作は $\{\beta^{13}, \dots, \beta, 1\}$ を基底とするベクトル空間に作用する行列として表現できる。 β を掛ける操作を $(2^{14}-1) - (561-14)$ 回行なう必要は無い。

【0050】【904】 $r(\beta)$ が訂正対象となる誤りパターンの形のいずれかになっているかを $\beta^{13}, \dots, \beta, 1$ を基底とするベクトル値で調べる。誤りパターンのいずれかであれば誤りイベントは $r(\beta)$ 、誤り位置は $r(\beta)$ を更新させた回数に応じて決まる(910)。この場合、誤りパターンと誤り位置が求められたので訂正可能。

【0051】【905】2コードワードにまたがる形の誤りイベント (図中では(*)を付けている) は符号ブロックの最上位か最下位の付近に位置するはずなので、誤り位置との整合性によって訂正可能(910)か訂正不能(909)かを判断する。

【0052】【906】 $r(\beta)$ が訂正対象となる誤りパターンに相当する値でなければ $r(\beta)$ に β をかけて $r(\beta)$ を更新する。

【0053】【907】更新回数が561回を上回るときは訂正対象外の誤りが発生したということで909のように訂正情報を設定する。

【0054】【908】訂正可能な場合は誤りを訂正する。その後、CRCCの冗長ビットを除く。

【0055】本実施例では、訂正対象となる誤りは符号

化ブロック内に1イベントのものである。符号化ブロック内に発生した2イベントの誤りについては、誤訂正を起こすものが存在する。図11は、2イベント誤りによる誤訂正を抑制するための方法の一例を示した図である。最尤復号器から得た尤度情報1101を用いて、尤度判定器1102は、誤りイベントの発生を推定する。尤度判定器1102が、誤りイベントの発生を推定したときはカウンタ1103の値を1増やす。しきい値判定器1104は、符号化ブロック内で推定誤りイベント数値が2以上か否かで、CRCCによる短い誤りの訂正8の動作のON/OFF切換えを行なう。

【0056】図10にCRCCを含む磁気ディスク装置(HDD)1000の構成の一例を示す。HDD1000は、磁気ディスク1002と磁気ヘッド1001とキャリッジ1003とキャリッジ上に取り付けられたR/W IC1004とスピンドルモータ1005、FPC1006等からなるヘッドディスクアセンブリ(HDA)1010、信号処理LSI(SPC)1011とハードディスクコントローラチップ(HDC)1012とサーボコントローラ(SRVC)1013、マイクロプロセッサ(MP)1014とSCSIチップ1015、ROM1016、RAM1017等からなるパッケージ基板(PCB)1030とて構成される。そして、短い誤り訂正を行なうための符号化と訂正を行なう回路CRCC8はSPCとHDCの間に位置する。

【0057】

【発明の効果】変調符号による0連続長制限を崩すことなく変調符号復調器10の直前に最尤復号器4で高頻度で発生する誤りを構成が簡単なCRCCを用いて訂正することで記憶チャネルの信頼性が向上される。

【図面の簡単な説明】

【図1】本発明の誤り訂正方式を組み込んだ信号処理ブロック構成図。

【図2】従来の信号処理のブロック構成図。

【図3】EPRMLの最尤復号における誤り流れ図。

【図4】線形誤り訂正符号のイメージ図。

【図5】誤り検出器の回路構成図。

【図6】16/17 (0,6/6) 符号において±(+---)の形の連続誤り長が最大になる例を示す図。

【図7】構成例における冗長ビットの構成図。

【図8】生成多項式とすることで長さ6の連続誤りでは誤訂正を起こさないような14次の原始多項式と、 E_1 に属する誤りでの誤訂正の有無の表。

【図9】CRCCにおける誤りのチェックと訂正を行なうアルゴリズムの流れ図。

【図10】CRCCを含む磁気ディスク装置の構成を示す図。

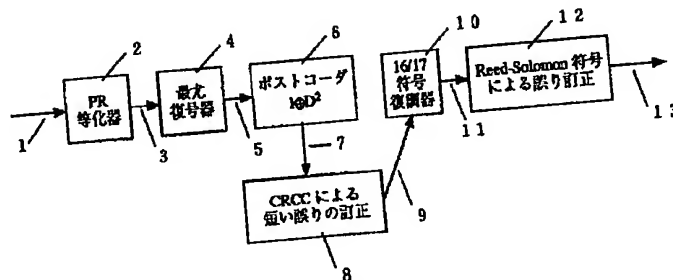
【図11】2イベント誤りによる誤訂正を抑制するための方法の一例を示した図。

【符号の説明】

1...ヘッドから読み出された信号、2...PR等化器、3...PR等化器によって等化された後の信号、4...最尤復号器、5...最尤復号器によって推定された符号系列、6...ポストコード化された推定符号系列、7...ポストコード化された推定符号系列、8...CRCCによる誤り訂正器、9...CRCCによる誤り訂正された後の推定符号系列、10...16/17符号復調器、11...推定符号系列に対応する情報系列、12...Reed-Solomon符号による誤り訂正器、13...Reed-Solomon符号によって誤り訂正された後の情報系列、501...CRCC誤り検出器、901~911...CRCCによる誤り検出・訂正アルゴリズムの説明に使用、1011...信号処理LSI(SPC)、1012...ハードディスクコントローラチップ(HDC)、1101...最尤復号器4から得られた尤度情報、1102...誤り発生尤度判定器、1103...推定誤りイベントのカウント、1104...誤り数のしきい値判定器。

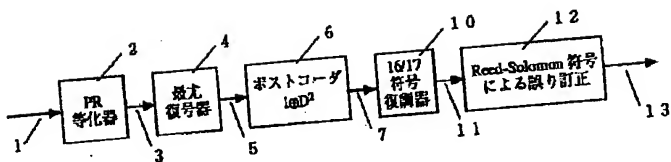
【図1】

図1



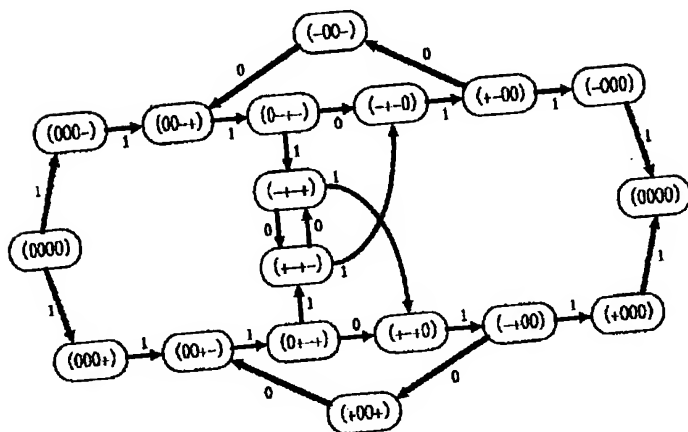
【図2】

図2



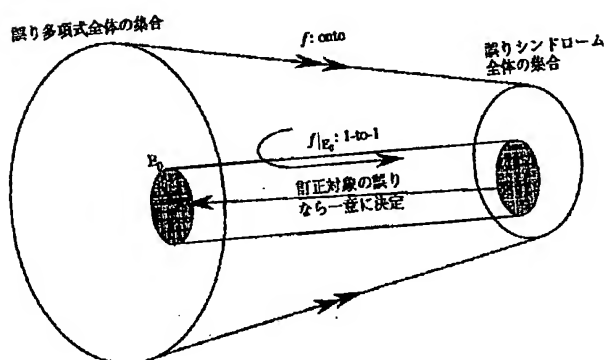
【図3】

図3



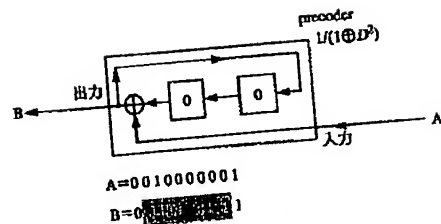
【図4】

図4



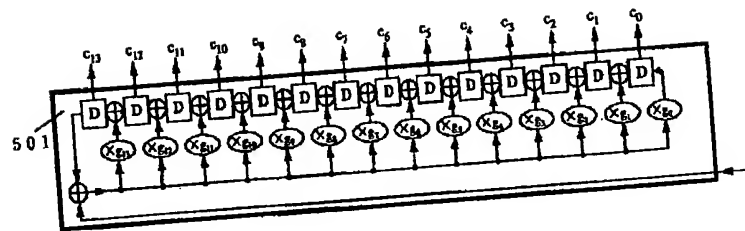
【図6】

図6



【図5】

図5



【図7】

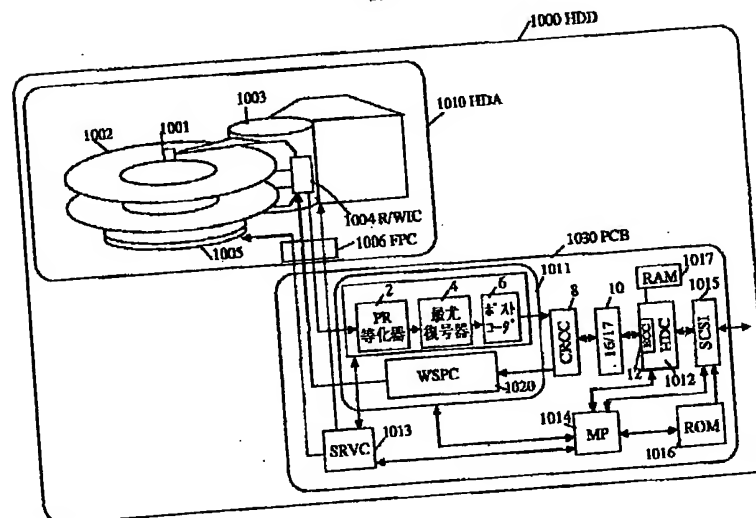
図7

生送多量式	0x4CA5	0x72CD	0x77D3
最大 符号での誤り	○	△	△
長さ6の 連続誤り	○	○	×
長さ7の 連続誤り	○	○	○
長さ8の 連続誤り	△	○	○
エラー率	×	○	○

- …誤訂正を起こさない
 △ …誤訂正が冗長ビットのみに
 起こるため、誤りが放って
 おかれたのと同様
 × …誤訂正によって誤りが増える

【図10】

図10



【図8】

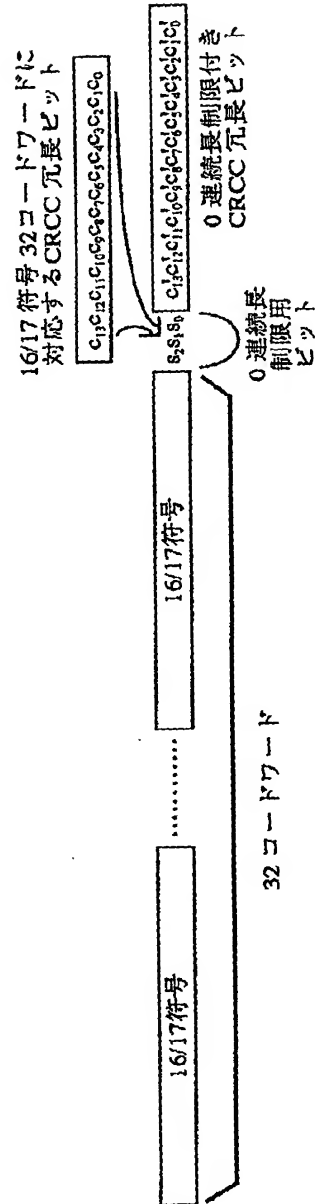
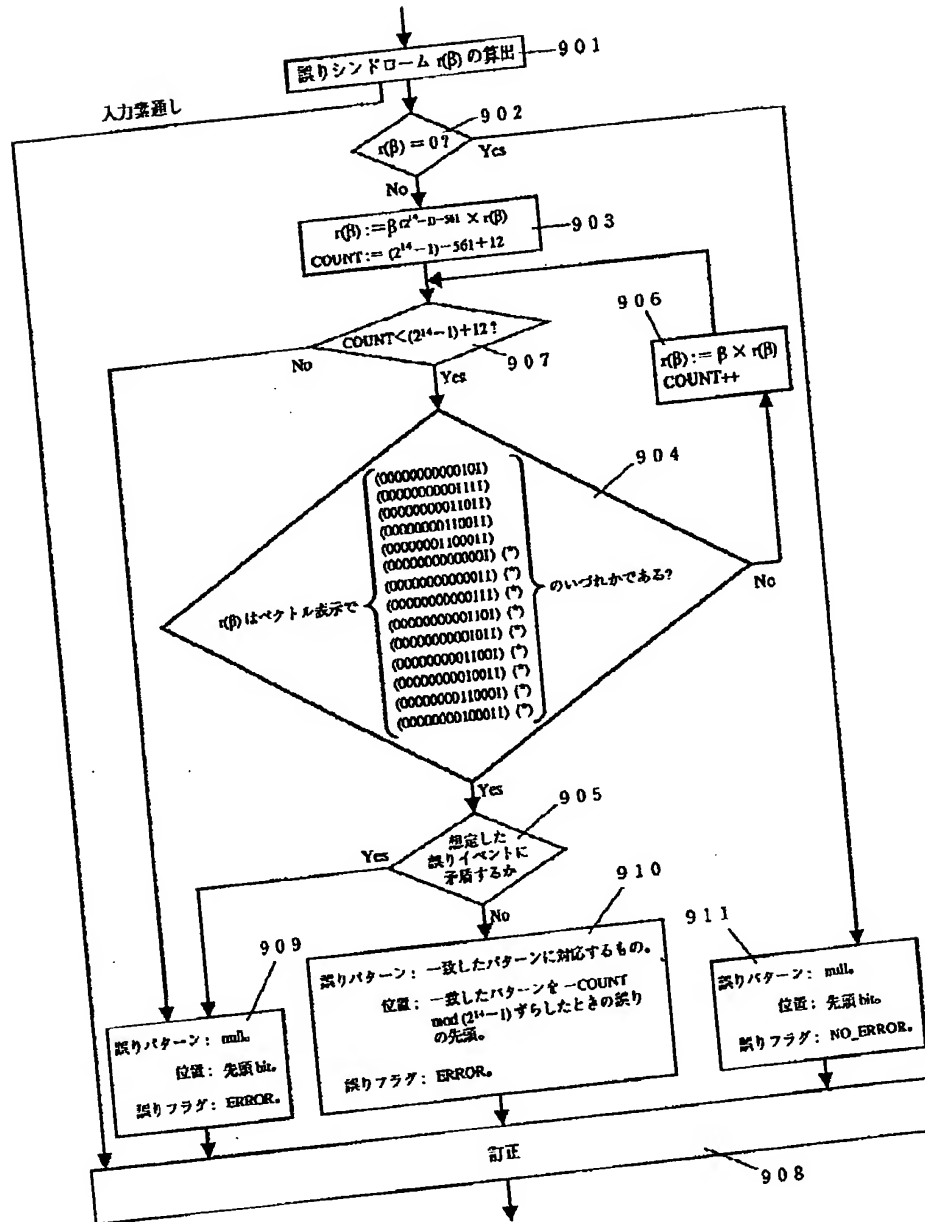


図 8

【図9】

図9



【図11】

図11

